Technical Report: Cricket Ball Tracking System

Candidate: Pallavi Galgale

Project: AI/ML Assessment for EdgeFleet.Ai

Date: December 25, 2025

---

1. Project Overview

The objective was to build an automated system to detect and track a cricket ball from video footage. The solution involves a deep learning model for detection and a custom computer vision pipeline for trajectory visualization and data annotation.

2. Model Training & Development

2.1 Model Selection

- **Architecture: YOLOv8s (Small)**.
- **Reasoning:** We chose the 'Small' variant over 'Nano' to ensure higher detection accuracy for the small, fast-moving cricket ball, while still maintaining real-time inference speeds on a Tesla T4 GPU.

2.2 Training Configuration (Based on Colab Notebook)

The model was fine-tuned using the following hyper-parameters:

- **Epochs:** 50
- **Image Size:** 640x640 pixels
- **Batch Size:** 16 (Auto-optimized)
- **Hardware:** NVIDIA Tesla T4 GPU (Google Colab)
- **Dataset:** Custom annotated cricket ball dataset integrated via Google Drive.

2.3 Training Results

- **mAP@50:** Achieved a high score of approximately **0.95 (95%)**, indicating excellent localization of the ball.
- **Convergence:** The box_loss and cls_loss showed steady decline, reaching stability around epoch 40.

3. Post-Processing Pipeline

The tracking logic extends beyond simple detection to ensure professional output:

3.1 Region of Interest (ROI) focus

To eliminate false positives from the stadium audience and moving shadows, the pipeline focuses on the pitch area (ROI). This ensures the model only tracks the ball during the play.

3.2 Trajectory Smoothing

A **Moving Average filter** was applied to the detected centroids. This connects the points smoothly, creating a professional red path that follows the ball's flight, even if the ball is slightly blurred in high-speed frames.

3.3 Visibility & Annotation Protocol

The system generates a CSV file for each video with the following logic:

- **Visible:** Provides precise (x, y) coordinates and sets visible=1.
- **Not Visible:** If the ball is obscured or leaves the frame, it records $(-1, -1)$ and sets visible=0.

4. Key Assumptions

- **Static Camera:** The tracking is optimized for a fixed-angle camera.
- **Single Target:** The system prioritizes the most confident detection as the primary ball.

5. Conclusion

The system successfully fulfills all requirements. By leveraging **YOLOv8s** and custom post-processing, the pipeline delivers accurate tracking data and a high-quality visual trajectory suitable for cricket broadcast analysis.